University of Málaga

Health Engineering

Assigment

Data mining

Author

Alejandro Domínguez Recio

Course

Intelligent Systems

Teachers

Enrique Domínguez Merino

Jesús de Benito Picazo

Introduction

In this assignment we are going to explain step by step the three algorithms studied for data mining (naive Bayes, ID3, k-means). They will be evaluated using a common dataset, with the most appropriate number of attributes and instances.

How are we going to do it?

To do the different studies we will have the following structure:

➔ **Dataset election**

In this part we are going to give a brief introduction to the context of the dataset, commenting on its type and the prediction objective.

➔ **Probabilistic classification**

We will do a detailed explanation of how the naive Bayes algorithm works and with what order it is created. We will also show the performance measures and discuss the results.

➔ **Decision tree**

We will do a detailed explanation of how the ID3 algorithm works and with what order it is created. We will obtain and comment on the same performance measures from the previous section by adding the size of the tree obtained or the number of leaves.

➔ **Clustering**

We will do a detailed explanation of how the k-means algorithm works and with what order it is created. We will also comment on its internal and external performance measures as well as the characteristics of the clusters obtained in comparison with the initial classes.

➔ **Permance measures for each* algorithm**

Accuracy

We can obtain the acurracy calculating the number of correctly predicted examples divided by the number of examples.

Precision

This measurement shows the positive predictive value, higher is bettter. We are going to obtain the necessary values from the confusion matrix and perform the following calculation TP/(TP+FP).

Fallout

This measurement shows the false positive rate, lower is better. We are going to obtain the necessary values from the confusion matrix and perform the following calculation FP/(FP+TN).

Recall

This measurement shows the true positive rate, higher is better. We are going to obtain the necessary values from the confusion matrix and perform the following calculation TP/(TP+FN).

F-measure

This measurement provides a single score that balances both the concers of precision and recall in one number, higher is better. We are going to obtain the necessary values of the previous measurements, precision and recall, and perform the following calculation 2*(Precision*Recall)/(Precision+Recall).

➔ **Specific measures of decision trees**

Tree size and number of leaves

We will create the tree in each step of the algorithm and once finished will obtain his measurements.

➔ **Specific measures of clustering**

Davies-Bouldin and Dunn

These measurements shows the compactness and separation of the clusters. We will use the mean and standard deviation of the differents clusters as well as the maximum distance between the samples of the clusters.

➔ **Evaluation of the results**

In this section we are going to compare the results of the different performance measures of each classification method.

➔ **Differences between algorithms**

Here we are going to describe the main differences both in the results and in the characteristics of the different algorithms.

➔ **Last conclusions**

Finally, we are going explain possible reasons why some algorithms shows better results than others. We will support our conclusions on the performance measures.

DATA MINING ALGORITHMS

➔ Dataset selection

we are going to work with a dataset with clinical data, specifically cardiac parameters. The prediction goal is given some input attributes to determine whether that person has heart disease or not. This dataset is a modification of the repository heart-statlog dataset provided by the University of Waikato for Weka. At first, this dataset consist only of real attributes which we will use for clustering, in the case of naive Bayes algorithms and decision trees they have been modified to categorical in order to work with them.

DATASET WITH REAL VALUES

Samples	X_1	X_2	X_3	X_4	Class
1	2.43	0.34	2.27	0.34	present
2	2.98	0.54	2.54	2.65	absent
3	1.22	0.33	0.90	2.83	present
4	2.07	0.67	0.36	0.92	absent
5	1.34	0.54	2.76	1.73	absent
6	2.64	0.09	0.45	2.83	absent
7	2.13	1.87	2.32	2.48	present
8	2.41	0.54	2.33	2.95	present
9	0.11	1.68	2.76	0.05	present
10	0.89	1.19	0.76	2.65	absent
11	1.67	0.72	1.44	1.38	present
12	1.33	0.54	1.87	1.18	absent
13	1.56	1.99	2.34	2.48	present
14	2.87	1.31	2.76	1.15	present
15	2.56	1.01	0.54	2.17	present
TEST SET					
16	1.45	0.25	2.84	2.12	present

17	0.78	0.37	0.54	1.49	absent
18	2.96	1.32	0.12	0.63	absent
19	1.58	1.81	0.78	0.71	absent
20	1.32	0.23	1.64	0.96	absent

DATASET WITH CATEGORICAL VALUES

Samples	X_1	X_2	X_3	X_4	Class
1	alto	no	alto	bajo	present
2	alto	no	alto	alto	absent
3	medio	no	bajo	alto	present
4	alto	no	bajo	bajo	absent
5	medio	no	alto	medio	absent
6	alto	no	bajo	alto	absent
7	alto	si	alto	alto	present
8	alto	no	alto	alto	present
9	bajo	si	alto	bajo	present
10	bajo	si	bajo	alto	absent
11	medio	no	medio	medio	present
12	medio	no	medio	medio	absent
13	medio	si	alto	alto	present
14	alto	si	alto	medio	present
15	alto	si	bajo	alto	present
TEST SET					
16	medio	no	alto	alto	present
17	bajo	no	bajo	medio	absent
18	alto	si	bajo	bajo	absent
19	medio	si	bajo	bajo	absent
20	medio	no	medio	bajo	present

How have we categorized the data?

Attribute	Numerical	Categorical
X_1,X_3,X_4	[0,00 0,99]	Bajo
	[1,00 1,99]	Medio
	[2,00 2,99]	Alto
X_2	[0,00 0,99]	Si
	[1,00 1,99]	No

- **Number of classes**

In this case our class attribute can take only **2 *values***, *absent* and *present*. Because of that we are faced to a ***binary classification problem***.

- **Number of attributes**

In this case we have **5 attributes** which are x_1 = *chest_pain { [0.00 2.99] [bajo medio alto] }*, x_2 = *fasting_blood_sugar { [0.00 1.99] [si no] }*, x_3 = *resting_electrocardiographic_results{ [0.00 2.99] [bajo medio alto]*, x_4 = *maximum_heart_rate_achieved { [0.00 2.99] [bajo medio alto] and class { absent, present }*

- **Number of samples**

In this case in particular the number of samples or instances is **15** as training set and **5** as test set**.**

➔Probabilistic classification

In relation to explain the operation of the naive Bayes classifier, we will number and explain the steps that are applied.

First of all, we are going to give a short introduction about the fundamentals of probabilistic classification and supervised learning.

Supervised Learning

Supervised learning is when you have input variables (x) and an output varible (y) and you train an algororithm to learn the mapping function from the input to the output.

- The number of posible outputs is named by range of f(x) and each element of the range is named a *class*.
- If the number of classes is 2 we are faced to a *binary classification problem*. Otherwise, it is multiclass.

Probablistic classifiers

- The probabilistic classifiers the output is the probability that given an input the output belongs to each of the classes.

With the data set provided above, we are going to develop the naive Bayes algorithm with estimation m = 4

I. Determine de number of classes and each element of these

Number of classes = C = 2

Range of C = V = { present, absent }

II. Determine our probability function

$$P(y|x)=\frac{P(y)*P(x|y)}{P(Present)*P(y|Present)+P(Absent)*P(y|Absent)}$$

$$P(x|y)=\prod_{d=1}^{4} P(x_d|y)$$

In our case we have 4 attributes so D = 4

III. The a priori class probabilities are obtained

Note

We determine our prior probability estimate, *p*, depending on the range of values of each attribute

$$P(Present)=\frac{ClassFrecuency}{TotalityOfsamples}=\frac{9}{15}=\frac{3}{5}$$

$$P(Absent)=\frac{ClassFrecuency}{TotalityOfsamples}=\frac{6}{15}=\frac{2}{5}$$

$$P(x_1|y)=\frac{n'+\frac{4}{3}}{n+4}$$

$$P(x_2|y)=\frac{n'+\frac{4}{2}}{n+4}$$

$$P(x_3|y)=\frac{n'+\frac{4}{3}}{n+4}$$

$$P(x_3|y)=\frac{n'+\frac{4}{3}}{n+4}$$

IV. Calculate the class probabilities for the test data x = {medio, no, alto, alto}

$$P(x|present)=P(x_1=medio|present)\,P(x_2=no|present)\,P(x_3=alto|present)\,P(x_4=alto|present)$$

$$P(x_1=medio|present)=\frac{3+\frac{4}{3}}{9}$$

$$P(x_2=no|present)=\frac{4+\frac{4}{2}}{13}$$

$$P(x_3=alto|present)=\frac{6+\frac{4}{3}}{12}$$

$$P(x_4=alto|present)=\frac{5+\frac{4}{3}}{12}$$

$$P(x|present)=\frac{2923}{1404}$$

$$P(x|absent)=P(x_1=medio|absent)\,P(x_2=no|absent)\,P(x_3=alto|absent)\,P(x_4=alto|absent)$$

$$P(x_1=medio|absent)=\frac{2+\frac{4}{3}}{9}$$

$$P(x_2=no|absent)=\frac{5+\frac{4}{2}}{13}$$

$$P(x_3=alto|absent)=\frac{2+\frac{4}{3}}{12}$$

$$P(x_4=alto|absent)=\frac{3+\frac{4}{3}}{12}$$

$$P(x|absent)=\frac{2173}{1404}$$

$$P(Present|x)=\frac{P(Present)*P(x|Present)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)}=0.6686$$

$$P(Absent|x)=\frac{P(Absent)*P(x|Absent)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)}=0.3313$$

We can conclude from the results that with said vector of attributes the predicted class is present

V. Calculate the class probabilities for the test data x = {bajo, no, bajo, medio}

$$P(x|present)=P(x_1=bajo|present)\,P(x_2=no|present)\,P(x_3=bajo|present)\,P(x_4=medio|present)$$

$$P(x_1=bajo|present)=\frac{1+\frac{4}{3}}{6}$$

$$P(x_2=no|present)=\frac{4+\frac{4}{2}}{13}$$

$$P(x_3=bajo|present)=\frac{2+\frac{4}{3}}{9}$$

$$P(x_4=medio|present)=\frac{2+\frac{4}{3}}{8}$$

$$P(x|present)=\frac{175}{6318}$$

$$P(x|absent)=P(x_1=bajo|absent)\,P(x_2=no|absent)\,P(x_3=bajo|absent)\,P(x_4=medio|absent)$$

$$P(x_1=bajo|absent)=\frac{1+\frac{4}{3}}{6}$$

$$P(x_2=no|absent)=\frac{5+\frac{4}{2}}{13}$$

$$P(x_3=bajo|absent)=\frac{3+\frac{4}{3}}{9}$$

$$P(x_4=medio|absent)=\frac{2+\frac{4}{3}}{8}$$

$$P(x|absent)=\frac{245}{5832}$$

$$P(Present|x)=\frac{P(Present)*P(x|Present)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)}=0.4972$$

$$P(Absent|x)=\frac{P(Absent)*P(x|Absent)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)}=0.5027$$

We can conclude from the results that with said vector of attributes the predicted class is absent

VI. Calculate the class probabilities for the test data x = {alto, si, bajo, bajo}

$$P(x|present)=P(x_1=alto|present)P(x_2=si|present)P(x_3=bajo|present)P(x_4=bajo|present)$$

$$P(x_1=alto|present)=\frac{6+\frac{4}{3}}{12}$$

$$P(x_2=si|present)=\frac{5+\frac{4}{2}}{10}$$

$$P(x_3=bajo|present)=\frac{2+\frac{4}{3}}{9}$$

$$P(x_4=bajo|present)=\frac{2+\frac{4}{3}}{7}$$

$$P(x|present)=\frac{55}{729}$$

$$P(x|absent)=P(x_1=alto|absent)P(x_2=si|absent)P(x_3=bajo|absent)P(x_4=bajo|absent)$$

$$P(x_1=alto|absent)=\frac{2+\frac{4}{3}}{12}$$

$$P(x_2=si|absent)=\frac{1+\frac{4}{2}}{10}$$

$$P(x_3=bajo|absent)=\frac{3+\frac{4}{3}}{9}$$

$$P(x_4=bajo|absent)=\frac{1+\frac{4}{3}}{7}$$

$$P(x|absent)=\frac{13}{972}$$

$$P(Present|x)=\frac{P(Present)*P(x|Present)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)}=0.8943$$

$$P(Absent|x)=\frac{P(Absent)*P(x|Absent)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)}=0.1056$$

We can conclude from the results that with said vector of attributes the predicted class is present

VII. Calculate the class probabilities for the test data x = {medio, si, bajo, bajo}

$$P(x|present)=P(x_1=medio|present)P(x_2=si|present)P(x_3=bajo|present)P(x_4=bajo|present)$$

$$P(x_1=medio|present)=\frac{3+\frac{4}{3}}{9}$$

$$P(x_2=si|present)=\frac{5+\frac{4}{2}}{10}$$

$$P(x_3=bajo|present)=\frac{2+\frac{4}{3}}{9}$$

$$P(x_4=bajo|present)=\frac{2+\frac{4}{3}}{7}$$

$$P(x|present)=\frac{169}{7290}$$

$$P(x|absent)=P(x_1=medio|absent)P(x_2=si|absent)P(x_3=bajo|absent)P(x_4=bajo|absent)$$

$$P(x_1 = medio|absent) = \frac{2+\frac{4}{3}}{9}$$

$$P(x_2 = si|absent) = \frac{1+\frac{4}{2}}{10}$$

$$P(x_3 = bajo|absent) = \frac{3+\frac{4}{3}}{9}$$

$$P(x_4 = bajo|absent) = \frac{1+\frac{4}{3}}{7}$$

$$P(x|absent) = \frac{13}{729}$$

$$P(Present|x) = \frac{P(Present)*P(x|Present)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)} = 0.2654$$

$$P(Absent|x) = \frac{P(Absent)*P(x|Absent)}{P(Present)*P(x|Present)+P(Absent)*P(x|Absent)} = 0.7346$$

We can conclude from the results that with said vector of attributes the predicted class is absent.

VIII. Calculate the class probabilities for the test data x = {medio, no, medio, bajo}

$$P(x|present) = P(x_1 = medio|present)P(x_2 = no|present)P(x_3 = medio|present)P(x_4 = bajo|present$$

$$P(x_1 = medio|present) = \frac{3+\frac{4}{3}}{9}$$

$$P(x_2 = no|present) = \frac{4+\frac{4}{2}}{13}$$

$$P(x_3 = medio|present) = \frac{1+\frac{4}{3}}{6}$$

$$P(x_4 = bajo|present) = \frac{2+\frac{4}{3}}{7}$$

$$P(x|present) = \frac{7}{243}$$

$$P(x|absent) = P(x_1 = medio|absent)\, P(x_2 = no|absent)\, P(x_3 = medio|absent)\, P(x_4 = bajo|absent)$$

$$P(x_1 = medio|absent) = \frac{2+\frac{4}{3}}{9}$$

$$P(x_2 = no|absent) = \frac{5+\frac{4}{2}}{13}$$

$$P(x_3 = medio|absent) = \frac{1+\frac{4}{3}}{6}$$

$$P(x_4 = bajo|absent) = \frac{1+\frac{4}{3}}{7}$$

$$P(x|absent) = \frac{245}{9477}$$

$$P(Present|x) = \frac{P(Present) * P(x|Present)}{P(Present) * P(x|Present) + P(Absent) * P(x|Absent)} = 0.3108$$

$$P(Absent|x) = \frac{P(Absent) * P(x|Absent)}{P(Present) * P(x|Present) + P(Absent) * P(x|Absent)} = 0.6892$$

We can conclude from the results that with said vector of attributes the predicted class is absent.

➔ **Values of permance measures**

Confusion Matrix

=== Confusion Matrix ===

a b ← classified as

1 1| a = present

1 2| b = absent

Accuracy

Correctly Classified Instances 3

Incorrectly Classified Instances 2

Total instances 5

Acurracy = 3/5 = **0,6%**

Precision

TP/(TP+FP) = 1/(1+1) = **0,5 %**

Fallout

FP/(FP+TN) = 1/(1+2) = **0,33 %**

Recall

TP/(TP+FN) = 1/(1+1) = **0,5 %**

F-measure

2*(Precision*Recall)/(Precision+Recall) = 2*(0,5*0,5)/(0,5+0,5) = **0.5**

➔ Section conclusion

In this section we have seen how the naive Bayes algorithm is implemented as well as the performance measures used in it. At the same time we have given a small introduction of what is supervised learning and probabilistic classifiers.

➔Decision tree

As in the previous section in order to detail the operation of the ID3 classifier, we will number and explain the steps that are applied.

First of all, we are going to give a short introduction about the fundamentals of decision trees.

Decision trees fundamentals

The decision trees belong to the family of supervised learning models. They are formed by roots, nodes and leaves. We use entropy as a measure of the uncertainty of the set of instances. In turn, we use the profit information to know how much uncertainty is reduced by choosing the division of a specific attribute.These models can be use for classification or regression problems. When the inputs are discrete we are faced to a classification problem.

We are going to develop an ID3 algorithm in the following points.

I. **We calculate the initial entropy**

$$H(x_1,\dots,x_{15})=0.9709$$

II. We calculate the gain in order to obtain the root node.

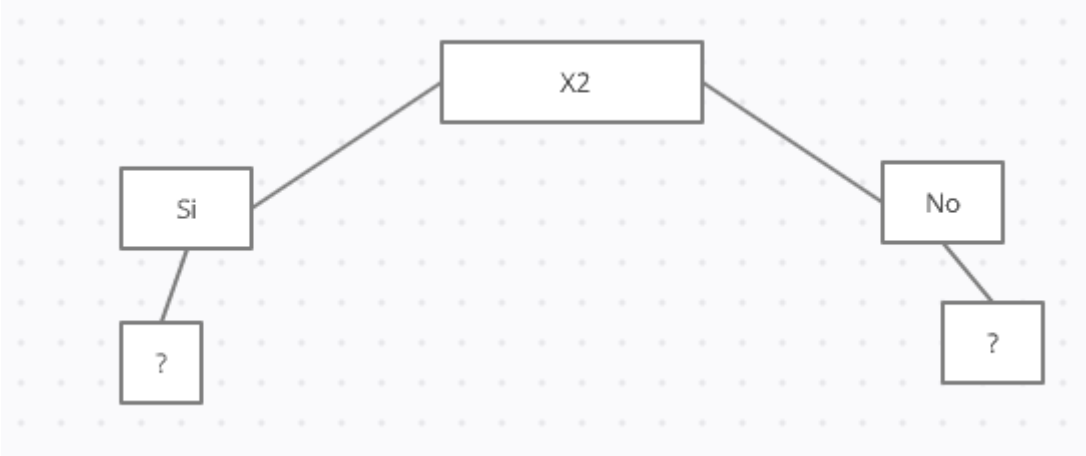
$$IG(s_1,\dots s_{15},X_1)=0.9709-\frac{2}{15}*1-\frac{5}{15}*0.9709-\frac{8}{15}*0.9544=0.00492$$

$$IG(s_1,\dots s_{15},X_2)=0.9709-\frac{6}{15}*0.6500-\frac{9}{15}*0.9910=0.1163$$

$$IG(s_1,\dots s_{15},X_3)=0.9709-\frac{5}{15}*0.9709-\frac{2}{15}*1-\frac{8}{15}*0.8112=0.0812$$

$$IG(s_1,...s_{15},X_4)=0.9709-\frac{3}{15}*0.9182-\frac{4}{15}*1-\frac{8}{15}*0.9544=0.0115$$

We choose as root node X2



We start by dividing then node X_2 = Si where S_{si} ={S_7, S_9, S_{10}, S_{13}, S_{14}, S_{15}}

Samples	X_1	X_2	X_3	X_4	Class
7	alto	si	alto	alto	present
9	bajo	si	alto	bajo	present
10	bajo	si	bajo	alto	absent
13	medio	si	alto	alto	present
14	alto	si	alto	medio	present
15	alto	si	bajo	alto	present

III. We calculate the initial entropy.

$$H(S_{Si}) = -((5/6)*\log 2(5/6) + (1/6)*\log 2(1/6)) = 0.6500$$

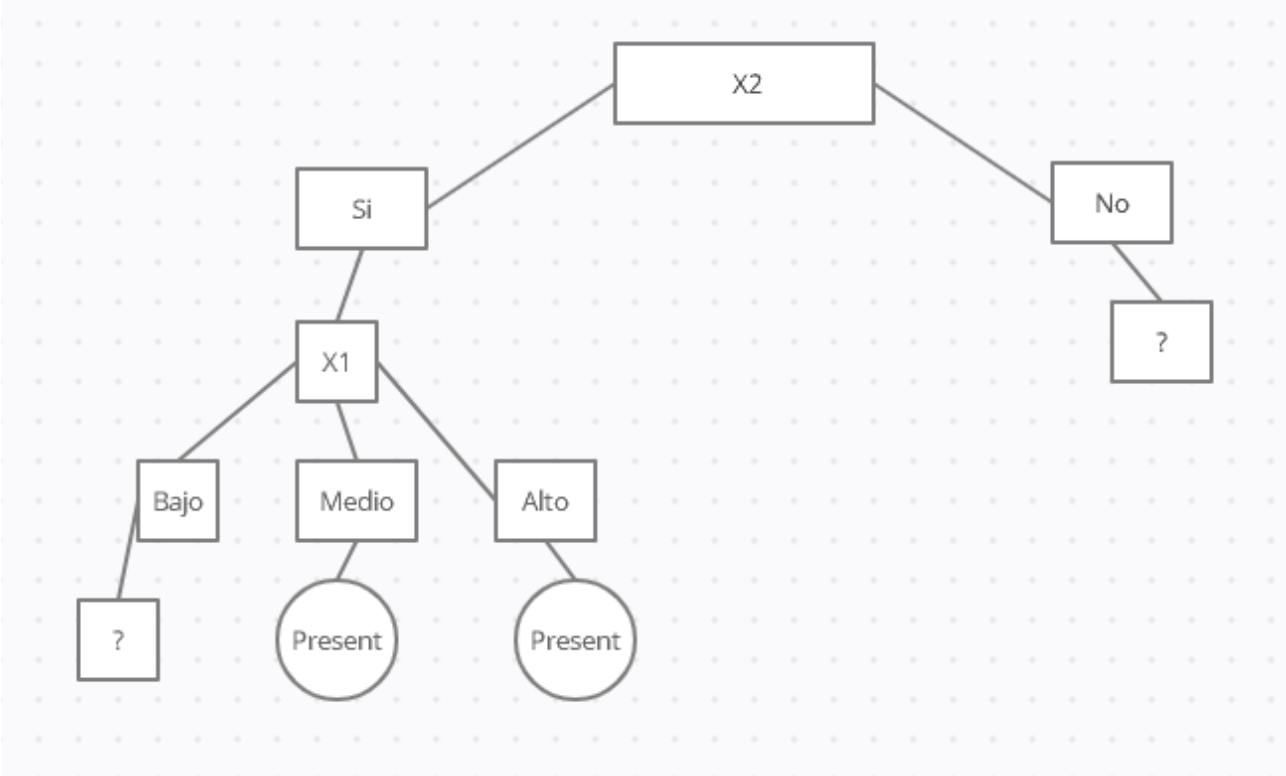
IV. We calculate IG in order to divide the node X_2 = Si

$$IG(S_{Si}, X_1) = 0.6500 - \frac{2}{6}*1 - \frac{1}{6}*0 - \frac{3}{6}*0 = 0.3166$$

$$IG(S_{Si}, X_3) = 0.6500 - \frac{2}{6}*1 - \frac{1}{6}*0 - \frac{3}{6}*0 = 0.3166$$

$$IG(S_{Si}, X_4) = 0.6500 - \frac{1}{6}*0 - \frac{1}{6}*0 - \frac{4}{6}*0.8112 = 0.1092$$

We can choose whether to divide the node X2 = Si between X1 and X3. If we choose X1 we have:



Now we must choose the attributes in which to divide the nodes X1 = Bajo

We start by dividing then node X1 = Si where $S_{Bajo} = \{S_9, S_{10}\}$

Samples	X_1	X_2	X_3	X_4	Class
9	bajo	si	alto	bajo	present
10	bajo	si	bajo	alto	absent

v. We calculate the initial entropy.

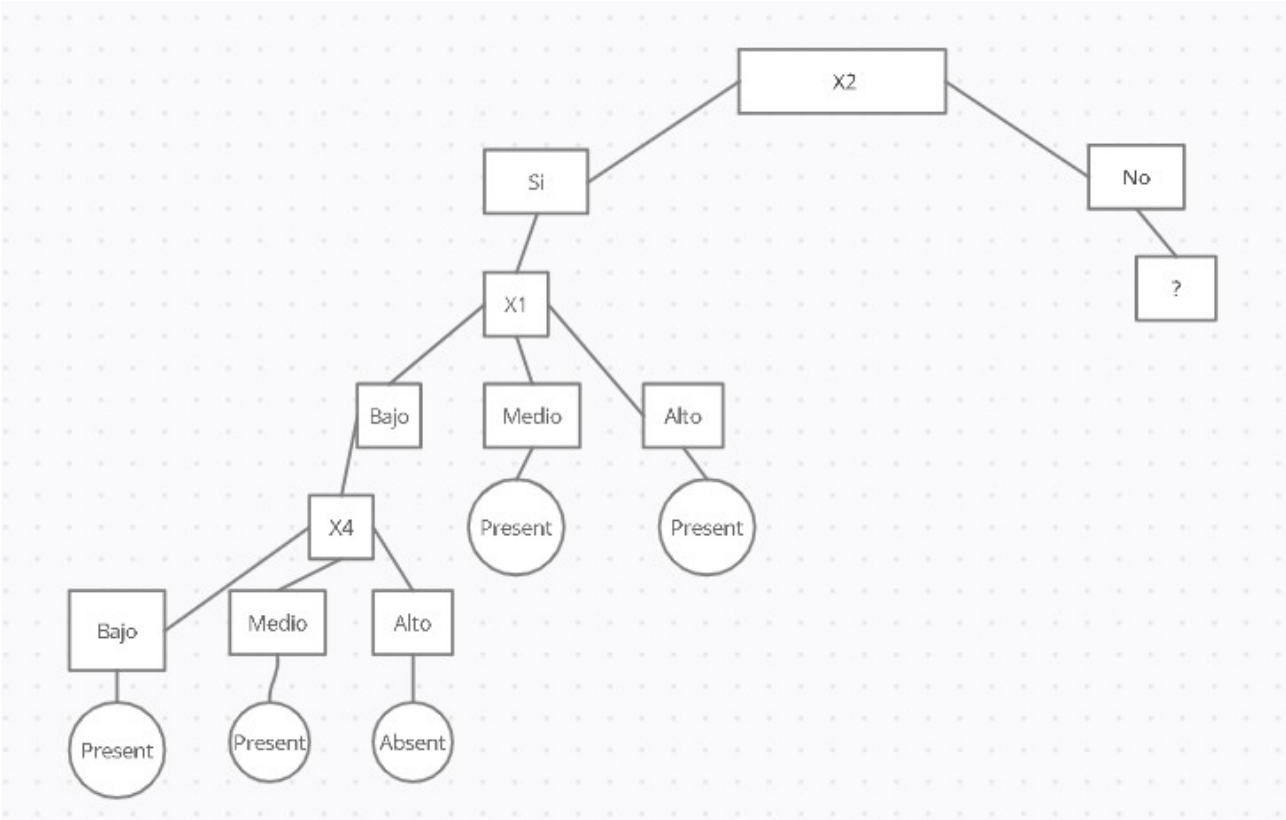
$$H(S_{Bajo}) = -((1/2) * \log 2(1/2) + (1/2) * \log 2(1/2)) = 1$$

VI. We calculate IG in order to divide the node X_1 = Bajo

$$IG(S_{Bajo}, X_3) = 1 - \frac{1}{2} * 0 - \frac{1}{2} * 0 = 1$$

$$IG(S_{Bajo}, X_4) = 1 - \frac{1}{2} * 0 - \frac{1}{2} * 0 = 1$$

We can choose whether to divide the node X1 = Bajo between X3 and X4. If we choose X4 we have no samples for X4 = medium. A possible solution to break this tie in the next



Now we must choose the attributes in which to divide the nodes X2 = no

We start by dividing then node X2 = No where $S_{No} = \{S_1, S_2, S_3 S_4, S_5, S_6, S_8, S_{11}, S_{12}\}$

Samples	X_1	X_2	X_3	X_4	Class
1	alto	no	alto	bajo	present
2	alto	no	alto	alto	absent
3	medio	no	bajo	alto	present
4	alto	no	bajo	bajo	absent
5	medio	no	alto	medio	absent
6	alto	no	bajo	alto	absent
8	alto	no	alto	alto	present
11	medio	no	medio	medio	present
12	medio	no	medio	medio	absent

VII. We calculate the initial entropy.

$$H(S_{No}) = -((4/9)*\log 2(4/9) + (5/9)*\log 2(5/9)) = 0.9910$$

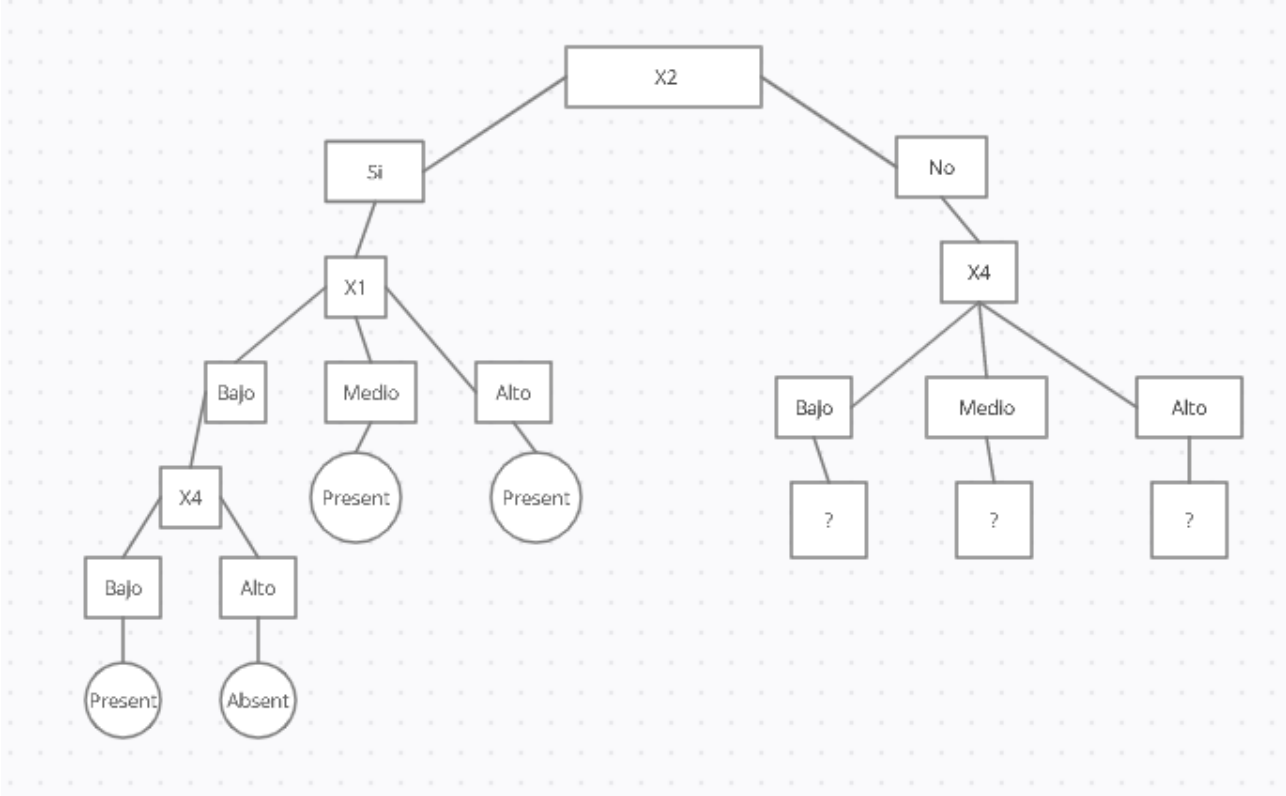
VIII. We calculate IG in order to divide the node X_2 = No

$$IG(S_{No}, X_1) = 0.9910 - \frac{4}{9}*1 - \frac{5}{9}*0.9709 = 0.0071$$

$$IG(S_{No}, X_3) = 0.9910 - \frac{3}{9}*0.9182 - \frac{2}{9}*1 - \frac{4}{9}*1 = 0.0182$$

$$IG(S_{No}, X_4) = 0.9910 - \frac{2}{9}*1 - \frac{3}{9}*0.9182 - \frac{4}{9}*1 = 0.0182$$

We can choose whether to divide the node X2 = No between X3 and X4. If we choose X4 we have:



Now we must choose the attributes in which to divide the nodes X4 = Bajo

We start by dividing node X4 = Bajo where $S_{No} = \{S_1, S_4,\}$

Samples	X_1	X_2	X_3	X_4	Class
1	alto	no	alto	bajo	present
4	alto	no	bajo	bajo	absent

IX. We calculate the initial entropy.

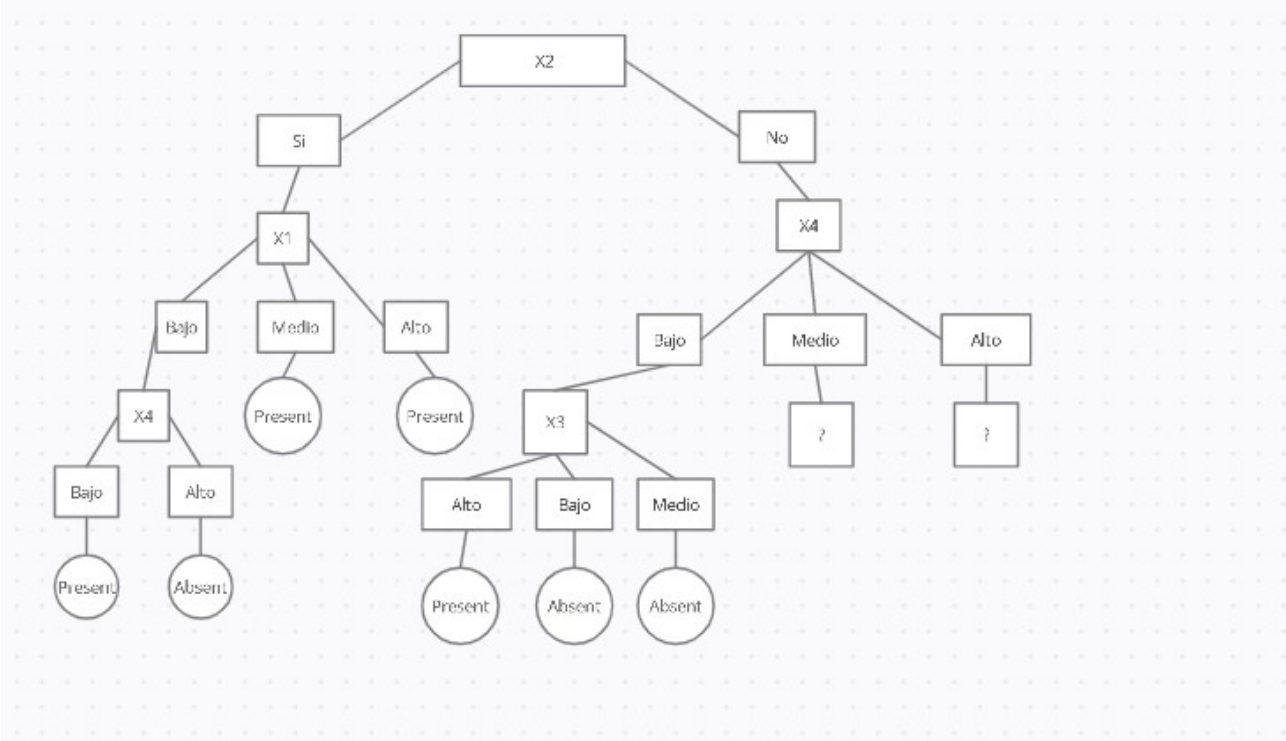
$$H(S_{Bajo}) = -((1/2)*\log 2(1/2) + (1/2)*\log 2(1/2)) = 1$$

X. We calculate IG in order to divide the node X_4 = Bajo

$$IG(S_{Bajo}, X_1) = 1 - \frac{2}{2} * 1 = 0$$

$$IG(S_{Bajo}, X_3) = 1 - \frac{1}{2} * 0 - \frac{1}{2} * 0 = 1$$

We choose X3 to divide the node X4 = Bajo but we have no samples for X3 = medio. A possible solution to break this tie in the next



Now we must choose the attributes in which to divide the nodes X4 = Medio

We start by dividing then node X4 = Medio where $S_{No} = \{S_5, S_{11}, S_{12}\}$

Samples	X_1	X_2	X_3	X_4	Class
5	medio	no	alto	medio	absent
11	medio	no	medio	medio	present
12	medio	no	medio	medio	absent

XI. We calculate the initial entropy.

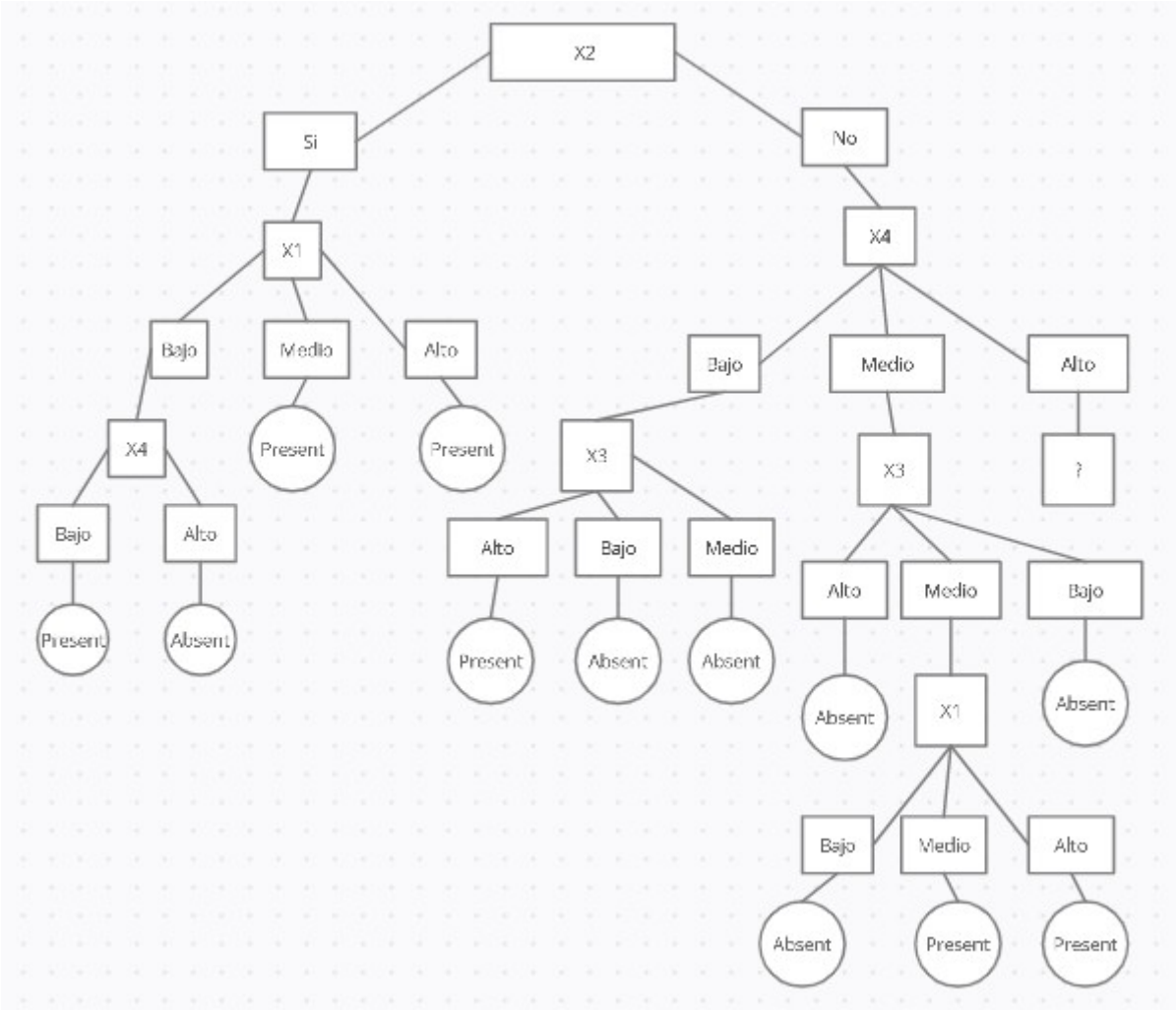
$$H(S_{Medio}) = -((2/3)*\log 2(2/3) + (1/3)*\log 2(1/3)) = 0.9182$$

XII. We calculate IG in order to divide the node X_4 = Medio

$$IG(S_{Medio}, X_1) = 0.9182 - \frac{3}{3}*0.9182 = 0$$

$$IG(S_{Medio}, X_3) = 0.9182 - \frac{1}{3}*0 - \frac{2}{3}*1 = 0.2515$$

We choose X3 to divide the node X4 = Medio but we have no samples for X3 = bajo. A possible solution to break this tie in the next. Also to divide X3 = medium we only have the attribute X1 having this only instances of X1 = medium so we are in the same situation as the previous one and we solve the tie in the same way.



We start by dividing then node X4 = Alto where $S_{No} = \{S_2, S_3, S_6, S_8\}$

Samples	X_1	X_2	X_3	X_4	Class
2	alto	no	alto	alto	absent
3	medio	no	bajo	alto	present

6	alto	no	bajo	alto	absent
8	alto	no	alto	alto	present

XIII. We calculate the initial entropy.

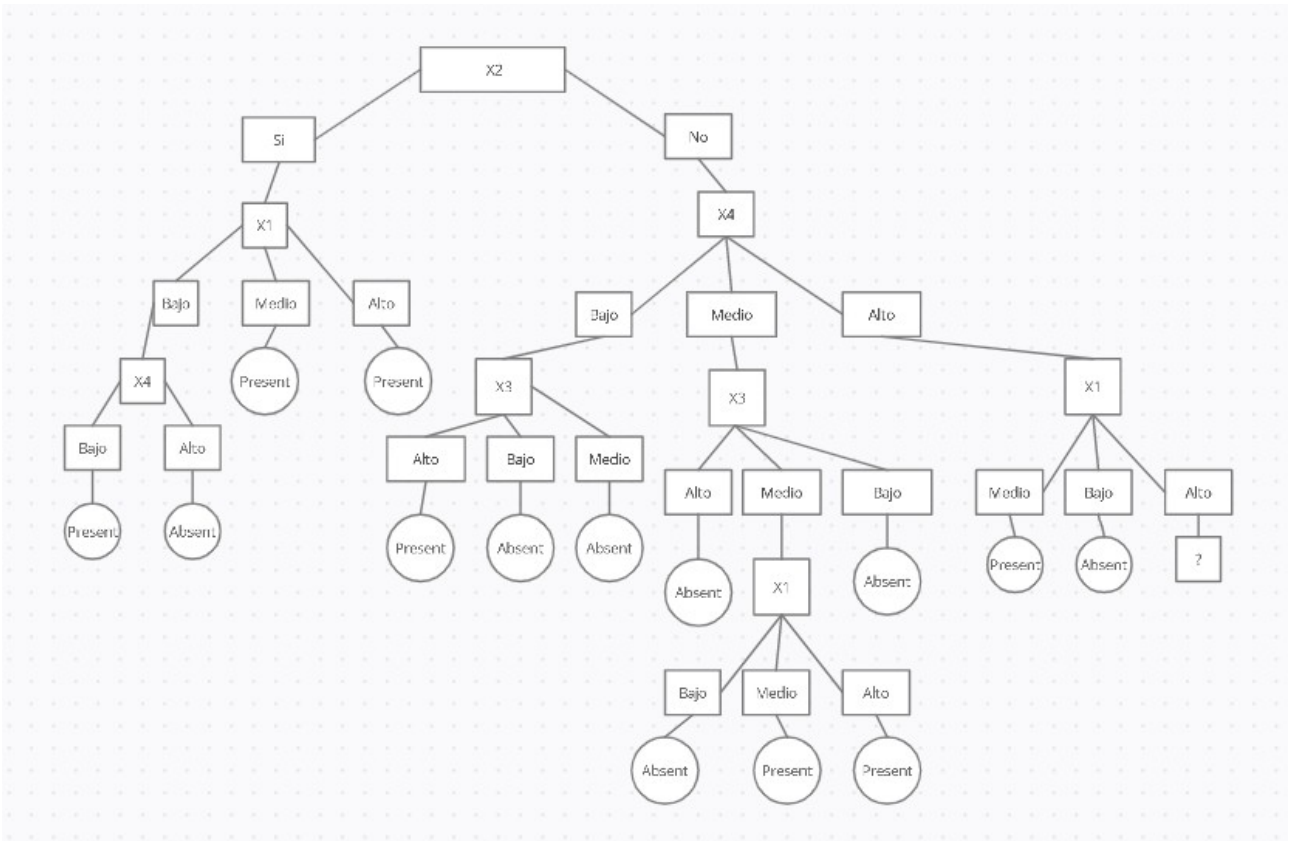
$$H(S_{Alto}) = -((2/4) * \log 2(2/4) + (2/4) * \log 2(2/4)) = 1$$

XIV. We calculate IG in order to divide the node X_4 = Alto

$$IG(S_{Alto}, X_1) = 1 - \frac{1}{4} * 0 - \frac{3}{4} * 0.9182 = 0.0818$$

$$IG(S_{Alto}, X_3) = 1 - \frac{2}{4} * 1 - \frac{2}{4} * 1 = 0$$

We choose X1 to divide the node X4 = Alto but we have no samples for X1 = bajo. A possible solution to break this tie in the next.

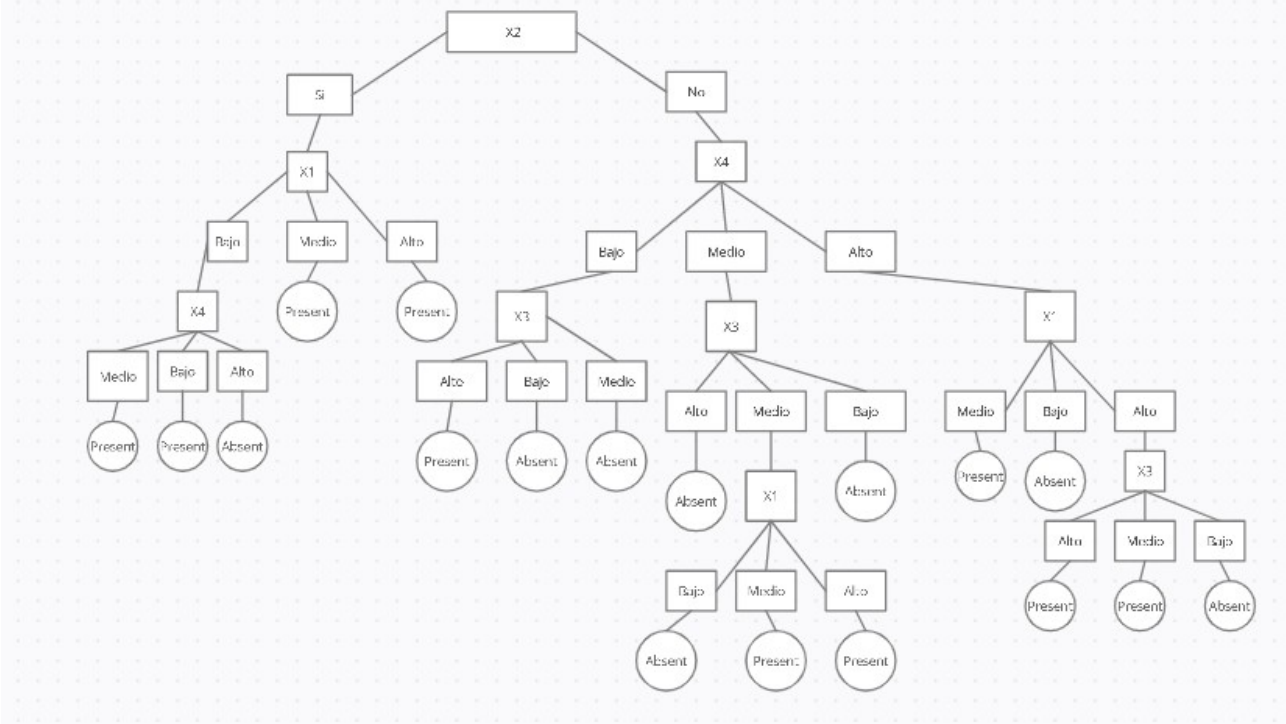


Now we must choose the attributes in which to divide the nodes X1 = Alto

We start by dividing then node X1 = Alto where $S_{No} = \{S_2, S_6, S_8\}$

Samples	X_1	X_2	X_3	X_4	Class
2	alto	no	alto	alto	absent
6	alto	no	bajo	alto	absent
8	alto	no	alto	alto	present

We have to take into account that in the branch X2 = No only do we have to go through X3 and in this case we have two types of classes alto and bajo but we have no samples for X3 = Medio. A possible solution to break this tie in the next.



Finally we are going to determine the prediction of classes given our test set

TEST SET						
Sample	X_1	X_2	X_3	X_4	Prediction	Class
16	medio	no	alto	alto	present	present
17	bajo	no	bajo	medio	absent	absent
18	alto	si	bajo	bajo	present	absent
19	medio	si	bajo	bajo	present	absent
20	medio	no	medio	bajo	absent	present

➔ **Values of permance measures**

Confusion Matrix

=== Confusion Matrix ===

a	b	← classified as
1	1	a = present
2	1	b = absent

Accuracy

Correctly Classified Instances 2

Incorrectly Classified Instances 3

Total instances 5

Acurracy = 2/5 = **40 %**

Precision

TP/(TP+FP) = 1/(1+2) = **0,33**

Fallout

FP/(FP+TN) = 2/(2+1) = **0,66**

Recall

TP/(TP+FN) = 1/(1+1) = **0,5**

F-measure

2*(Precision*Recall)/(Precision+Recall) = 2*(0,33*0,5)/(0,33+0,5) = **0,397**

Tree size and number of leaves

Leaves = **18** Size = **49**

➔ Section conclusion

In this section we have developed the ID3 algorithm as the decision tree resulting from it. At the same time we have verified some of its disadvantages in cases of having a disproportionate data set and the need to create labels that are not specified in the original data set with the reliability problems that this entails. I would also add the ease of understanding that this has once the model is created and the advantages that this entails when explaining the reason for its conclusions.

➔Clustering

As in the previous sections in order to detail the operation of the K-means classifier, we will number and explain the steps that are applied.

We are going to give a short introduction about the fundamentals of clustering.

Fundamentals of clustering

In the first place, we can define clustering as the technique of finding groupings within a data set without an associated class or label in such a way that the data or objects grouped within the same cluster are similar to each other and different from the objects of other groups.

As we have said before, it is an unsupervised learning technique in which we do not have predefined classes. Within the different clustering algorithms we can differentiate hierarchical and non-hierarchical within this last group, we highlight K-means. To measure the distance between the points of the different groups or within these, various methods or techniques such as Euclidean or Manhattan, among others, can be used. The evaluation of the performance or reliability of the clusters obtained we have internal and external measures that generally determine how compact and separate the clusters are between them or the points within them.

We will apply the algorithm with the following starting points assuming we have two groups. In turn, to calculate the distances we will use Euclidean squared.

M1	*1.00*	*0.50*	*2.00*	*1.50*
M2	*1.25*	*1.00*	*2.50*	*0.75*

I. We calculate the sum of the distances to the centroids.

Instances/Centroids	M1	M2
1	3,489	**2,049**
2	**5,5132**	6,7782
3	**3,0562**	7,3362
4	**4,1998**	5,3898
5	**0,7477**	1,2477
6	**7,0291**	11,2891
7	**4,2166**	4,5566
8	**4,2011**	6,4261
9	4,8646	**2,3196**
10	**3,3483**	6,8033
11	**0,8253**	1,7753
12	**0,2298**	0,7998
13	**3,6097**	4,0947
14	4,8531	**2,9481**
15	**5,2742**	7,5742

How is the process to calculate the distances?

As we have said, we use the Euclidean squared distance as a method to calculate the distances between the instances and the centroids.

$$\|x - y\| = \sum_{i=1}^{d} (x_i - y_i)^2$$

And the process used is the following:

Samples	X_1	X_2	X_3	X_4	Class
1	2.43	0.34	2.27	0.34	present

Here we can see the data of our first instance with its four attributes ($X_{1,..}$ $,X_4)$ and their respective positions.

M1	*1.00*	*0.50*	*2.00*	*1.50*

And here we have the positions ($Y_{1,...,}Y_{4b}$) of the centrioles associated with the first cluster.

Therefore incorporating these values to our chosen distance measure we have the <u>distance from the first instance the centroids of the cluster 1</u>

Instance1/Cluster1 =

$$(2{,}43-1{,}00)^2+(0{,}34-0.50)^2+(2{,}27-2{,}00)^2+(0{,}34-1{,}50)^2=3.489$$

And so with each instance for each cluster in each iteration

We update the centroids

M1	*1.9000*	*0.8358*	*1.5508*	*2,1866*
M2	*1.8033*	*1.11*	*2.5966*	*0.5133*

How is the process to calculate the centroids?

The choice of centroids is done randomly in the first place, but in subsequent iterations we follow the following process:

1)We associate the instances to each cluster according to the result of the sum of their points to the previous centroids, ordering the choice by minimum length.

Instances/Centroids	M1	M2
1	3,489	**2,049**
2	**5,5132**	6,7782
3	**3,0562**	7,3362
4	**4,1998**	5,3898

For example in these four instances we will group M2 = [S_1] and M3 = [S_2,S_3,S_4] and so on with the rest of samples.

Once we have all the instances grouped to their closest cluster, we obtain the positions of the new centroids by performing the calculation shown in the following example:

$M2/X_{1} = \frac{S_1+S_9+S_{14}}{3} = 1,8033$

I. Calculate the sum of the distances to the centroids.

Instances/Centroids	M1	M2
1	10,0786	**5,1309**

2	**9,3916**	11,2748
3	**4,5951**	11,5095
4	**8,1552**	9,3412
5	**5,5348**	4,8118
6	**8,4908**	15,5677
7	**5,9627**	8,2367
8	**7,2780**	10,9504
9	10,1640	**4,4932**
10	**4,4530**	11,2388
11	**4,9687**	5,6543
12	4,9693	**4,2727**
13	**4,8433**	7,3881
14	10,2846	**6,7815**
15	**7,6048**	12,3286

We update the centroids

M1	*1,9518*	*0,8627*	*1,5218*	*2,2781*
M2	*1,6850*	*0,9675*	*2,4150*	*0,6800*

II. Calculate the sum of the distances to the centroids.

Instances/Centroids	M1	M2
1	10,5819	**5,1147**
2	**9,4601**	10,7466
3	**4,6284**	9,8207
4	8,4613	**8,3353**
5	5,8708	**4,1995**
6	**8,4216**	14,1202
7	**6,0934**	7,4396
8	**7,3082**	10,0801
9	10,8736	**4,0904**
10	**4,5289**	9,4651

11	5,2745	**4,7733**
12	5,3549	**3,5176**
13	**5,0059**	6,4359
14	10,6445	**6,7986**
15	**7,6659**	11,1110

We update the centroids

M1	*2,0487*	*0,9450*	*1,5225*	*2,6287*
M2	*1,6885*	*0,8285*	*2,0314*	*0,9642*

III. Calculate the sum of the distances to the centroids.

Instances/Centroids	M1	M2
1	12,1993	**5,8623**
2	**9,3738**	10,5469
3	**4,7008**	7,7655
4	9,7338	**7,1878**
5	6,6922	**4,2299**
6	**8,2609**	12,1146
7	**6,2595**	6,9258
8	**7,0986**	9,3847
9	13,0754	**4,7338**
10	**4,7820**	7,3131
11	6,2888	**4,0817**
12	6,5648	**3,1751**
13	**5,2658**	5,7790
14	11,6207	**7,5841**
15	**7,9811**	9,5273

Because the group assignment is the same as in the previous section, we stop the execution of the algorithm.

Now to obtain external performance measures such as cluster similarity or the number of hits we use the test data set

Instances/Centroids	M1	M2
16	2,8360	**2,3812**
17	4,2021	**3,5363**
18	6,9329	**5,6234**
19	5,2006	**2,6057**
20	3,8405	**0,6472**

➔ **Values of permance measures**

➔ Internal Evaluation

These performance measures tell us how the clusters are compacted and the separation between them.

Davies-Bouldin and Dunn

$$DB = \frac{1}{k}\sum_{i=1}^{k} \max_{j \neq i}\left(\frac{\sigma_i + \sigma_j}{d(\mu_i, \mu_j)}\right)$$

$$D = \min_{i \in \{1,\ldots,k\}}\left(\min_{j \neq i}\left(\frac{d(\mu_i, \mu_j)}{\max_{h \in \{1,\ldots,k\}} \Delta_h}\right)\right)$$

For the internal performance measures of Davies Bouldin and Dunn we will need

-The mean vector of cluster i.

We use the centroids obtained from the final clustering

M1	*2,0487*	*0,9450*	*1,5225*	*2,6287*
M2	*1,6885*	*0,8285*	*2,0314*	*0,9642*

-The standard deviation of cluster i.

To calculate the standard deviation we are going to follow the following formula

$$\sigma = \frac{\sum_i^N |X_i - \bar{X}|}{N}$$

We obtain the data from the table of sum of the distances of the last iteration.

Instances/Centroids	M1	M2
1	12,1993	**5,8623**
2	**9,3738**	10,5469
3	**4,7008**	7,7655
4	9,7338	**7,1878**
5	6,6922	**4,2299**
6	**8,2609**	12,1146
7	**6,2595**	6,9258
8	**7,0986**	9,3847
9	13,0754	**4,7338**
10	**4,7820**	7,3131
11	6,2888	**4,0817**
12	6,5648	**3,1751**
13	**5,2658**	5,7790
14	11,6207	**7,5841**
15	**7,9811**	9,5273
$TotalSumCluster_i$	**53,7225**	**36,8547**
InstancesPerCluster	**8**	**7**

Therefore we have:

Standard deviation of M1 = 58,7225/8 = 6,7215

Standard deviation of M2 = 36,8547/7 = 5,2649

-The maximun distance among samples of cluster i.

To calculate the maximum distance between two samples we will use the squared Euclidean distance as we have been doing in the rest of the sections. In our case, the maximum distance is between $S_2$and S_9, being this:

$$(2.98-0.11)^2+(0.54-1.68)^2+(2.54-2.76)^2+(2.65-0.05)^2=16,293$$

Once we have all the necessary values we proceed to perform the calculations:

$$DB = \frac{6,7215+5,2649}{3,1728} = 3,7778$$

$$D = \frac{3,1728}{16,2930} = 0,1947$$

➔ *External evaluation*

To determine an external evaluation of our clustering, we use the class information associated with each sample of our test data set.

Instances/Centroids	M1	M2
16	2,8360	**2,3812**
17	4,2021	**3,5363**
18	6,9329	**5,6234**
19	5,2006	**2,6057**
20	3,8405	**0,6472**

As we can see, only one cluster has been formed with our test data set. In this we had two groups, the largest of them being 3 instances, so the result of our test has at least 2 incorrectly clustered instances and a maximum of 3. If we associate the M2 cluster to the absent class we have:

Confusion Matrix

=== Confusion Matrix ===

a b ← classified as

0 2| a = Present

0 3| b = Absent

Accuracy

Correctly Classified Instances 3

Incorrectly Classified Instances 2

Total instances 5

Acurracy = 3/5 = **0,6**

➔ **Section conclusion**

In this section we have developed the K-means algorithm with squared Euclidean distance. Regarding the characteristics of this I can highlight the ease of incorporating the test data as well as the ease of implementation, on the other hand the choice of the number of clusters or the centroids is something that can have a great impact on the quality of our predictions and that requires a previous study for its optimization.

➔ Evaluation of the results

Algorithm	Acurracy	Precision	Fallout	Recall	F-measure
Naive Bayes	0,6	0,5	0,33	0,5	0,5
ID3	0,4	0,33	0,66	0,5	0,397
K-means	0,6				

➔ Differences between algorithms

Among the first differences that we can highlight is a better result of naive Bayes in performance measures with respect to the rest of classifiers, although in comparison with k-means in accuracy it has the same value in the rest, k-means does not provide null values for the choice of positive and negative classes in the confusion matrix. In turn, it has been observed that ID3 has had some node divisions in which there were not enough classes and it has had to opt for a choice not supported by profit values with the effect that this has on the result of its predictions. On the other hand, when applying the test data set to k-means, the values from said test points towards the centroids have not had disproportionate values compared to the range of values between the centroids of the clusters and the points of the training data set. This could have required starting the algorithm again to update the centroids and the corresponding clusters, it would also add that in k-means the change from categorical to numerical values has been able to create a disproportion in the data with the consequent effect on the results.

➔ Last conclusions

By way of conclusion I would like to highlight that I have learned with the classification algorithms, the differences between them and with the experience that we have obtained when using each one of them.

First of all, I have learned what classification algorithms and supervised and unsupervised learning are. We can use supervised learning when we have the information tagged, being able to provide this information to the algorithm so that it can classify it according to it and thus make its predictions. Regarding the types of data that we can use we have numerical and categorical, being the choice of the algorithm a decision that can be conditioned by the type of these, for example naive Bayes has a better performance with categorical data which has conditioned than in the examples carried out, this type of data has been provided. With respect to unsupervised learning, in which, unlike supervised learning, our data do not have labels or classes associated with which to identify and classify them, so the algorithms that work in this field are in charge of classifying or grouping said data in clusters. or groups being a way of categorizing them.

Among the main differences that we have observed, we can group them into performance differences, which we obtain from the measures studied such as accuracy, precision, ROC,. . . , implementation differences or testing differences. Starting with the performance measures, we have learned measures that are common for the three types of classifiers studied, such as accuracy, precision, fallout, recall or ROC, although in the case of clustering we need our data or clusters to be labeled in order to obtain the necessary values for such measures. In relation to these we can say that we have to have a critical sense when evaluating and deciding without an algorithm it performs well or not for a specific data set since poor quality or unbalanced training data can give values that are not adjust to later tests with other

datasets. The ROC measurement may be the most reliable even with unbalanced datasets. We also highlight 'exclusive' measures of the some algorithms such as the measures of tree size or number of leaves in decision or cohesion in k-means. Among the differences in implementation we highlight the ease of understanding of the decision trees, the simplicity of k-means or how with some basic notions of probability we can apply naive Bayes. Once implemented between the testing differences we can highlight the speed and ease of understanding of a decision tree prediction, on the contrary, in naive Bayes it is necessary to carry out the operations of the algorithm itself and in k-means it is necessary to calculate the distances of the set points to the already clustered centroids as well as a possible re-evaluation of the algorithm to update the centroids in case the new test data deviates above a threshold value.